IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICANT(S): NOBUYUKI KIHARA AND TEPPEI YOKOTA

SERIAL NO.: 09/535,003 GROUP ART UNIT: 2131

FILED: March 24, 2000 EXAMINER: SHERKAT, AREZOO

INVENTION: EDITING APPARATUS AND EDITING METHOD

Hon. Commissioner of Patents and Trademarks,
Washington, D.C. 20231

SIR:

CERTIFIED TRANSLATION

Yuka NAKAMURA residing at c/o SUGIURA PATENT OFFICE, 7th floor, Ikebukuro Park Bldg., 49-7, Minami Ikebukuro 2-chome, Toshima-ku, Tokyo, JAPAN, declares:

(1) that she knows well both the Japanese and English languages;

(2) that she translated Japanese Application 11-081535 from Japanese to English;

(3) that the attached English translation is a true and correct translation of the above-identified Japanese Application to the best of her knowledge and belief; and

(4) that all statements made of her own knowledge are true and that all statements made on information and belief are believed to be true, and further that these statements are made with the knowledge that willful false statements and the like are punishable by fine or imprisonment, or both, under 18 USC 1001, and that such false statements may jeopardize the validity of the application or any patent issuing thereon.

May 17, 2004 | Yuka Nakamura
Date | Yuka NAKAMURA

[Title of Document] Specification

[Title of the Invention] Data processing Apparatus and Method

[Scope of Claims for a Patent]

[Claim 1]

A data processing apparatus having an attachable/detachable non-volatile file memory from which data is readable by a computer, comprising:

a controlling portion for handling a data file and file management information for managing the data file; and

a memory interface disposed between said controlling portion and the non-volatile memory,

wherein the file management information is composed of:

a first file information area containing at least the file name of the data file and the number of parts that compose the data file, the number of parts being at least one; and

a second file information area containing at least the size of each of the part, the part start position, and the part end position, and

wherein when the data file is divided into two data files, the data processing apparatus is adapted for updating the size and the file end position of the second file information area corresponding to the data file so as to generate the file management

information corresponding to a first data file placed before a divide point of the data file and for copying a predetermined area containing the divide point and placing the copied predetermined area at the beginning of the second data file so as to generate the file management information corresponding to a second data file placed after the dividing point.

[Claim 2]

A data processing apparatus having an attachable/detachable non-volatile file memory from which data is readable by a computer, comprising:

a controlling portion for handling a data file and file management information for managing the data file; and

a memory interface disposed between said controlling portion and the non-volatile memory,

wherein the file management information is composed of:

a first file information area containing at least the file name of the data file and the number of parts that compose the data file, the number of parts being at least one; and

a second file information area containing at least the size of each of the part, the part start position, and the part end position, and

wherein when a first data file and a second data file are combined, the data processing apparatus

is adapted for successively placing the first file information area and the second file information area corresponding to the first data file and the second file information corresponding to the second data file so as to generate the file management information corresponding to the combined data file.

[Claim 3]

A data processing method for a data processing apparatus having an attachable/detachable non-volatile file memory from which data is readable by a computer,

the non-volatile memory storing a data file and file management information for managing the data file,

the file management information being composed of a first file information area containing at least the file name of the data file and the number of parts that compose the data file, the number of parts being at least one, and a second file information area containing at least the size of each of the part, the part start position, and the part end position,

wherein the data file is divided into two data files, the data processing method comprising the steps of:

updating the size and the file end position of the second file information area corresponding to the data file so as to generate the file management

information corresponding to a first data file placed before a divide point of the data file; and

copying a predetermined area containing the divide point and placing the copied predetermined area at the beginning of the second data file so as to generate the file management information corresponding to a second data file placed after the dividing point.

[Claim 4]

A data processing method for a data processing apparatus having an attachable/detachable non-volatile file memory from which data is readable by a computer,

the non-volatile memory storing a data file and file management information for managing the data file,

the file management information being composed of a first file information area containing at least the file name of the data file and the number of parts that compose the data file, the number of parts being at least one, and a second file information area containing at least the size of each of the part, the part start position, and the part end position,

wherein a first data file and a second data file are combined, the data processing method comprising the step of:

successively placing the first file information area and the second file information area

corresponding to the first data file and the second file information corresponding to the second data file so as to generate the file management information corresponding to the combined data file.

[Detailed Description of the Invention]

[0001]

[Technical Field to which the Invention belongs]

The present invention relates to a suitable data processing apparatus and method for applying the detachable memory card to the apparatus as the record medium such as the audio data.

[0002]

EEPROM (Electrically Erasable Programmable ROM) that is an electrically rewritable memory requires a large space because each bit is composed of two transistors. Thus, the integration of EEPROM is restricted. To solve this problem, a flash memory that allows one bit to be accomplished with one transistor using all-bit-erase system has been developed. The flash memory is being expected as a successor of conventional record mediums such as magnetic disks and optical discs.

[0003]

A memory card using a flash memory is also known. The memory card can be freely attached to an apparatus and detached therefrom. A digital audio recording/reproducing apparatus that uses a memory card

instead of a conventional CD (Compact Disc) or MD (Mini Disc) can be accomplished.

[0004]

A file management system used for a conventional personal computer is named FAT (File Allocation Table). In the FAT system, when a particular file is defined, predetermined parameters are successively set to the file. Thus, the size of a file becomes variable. One file is composed of at least one management unit (sector, cluster, or the like). Data corresponding to the management unit is written to a table referred to as FAT. In the FAT file system, a file structure can be easily formed regardless of the physical characteristics of a record medium. Thus, the FAT file system can be used for a magneto-optical disc as well as a floppy disk and a hard disk. In the above-mentioned memory card, the FAT file system is used.

[0005]

However, a CD with which audio data is recorded does not have the concept of the FAT system at all. In the era of the MD with which audio data can be recorded and reproduced, music programs are recorded and edited using an FAT modified system called Link-P. Thus, the system itself can be controlled with a low power CPU. However, using such a system, data cannot be exchanged with a personal computer. Thus, the MD

system has been developed as an isolated AV system.

[0006]

The link-P system used in the MD is composed of a P-DFA (Pointer for Defective Area), a P-Empty (Pointer for Empty slot) area, a P-FRA (Pointer for FReely Area), and P-TNO1 ... P-TNo255. The P-DFA represents the top position of a slot that contains information of a defect on an MD. The P-Empty area represents the use state of a slot. The P-FRA represents the top position of a slot used for managing a recordable area. The P-TNo1, P-TNo2, ..., P-TNo255 represent the start positions of slots corresponding to individual music programs.

[0007]

Next, with reference to Fig. 29, a process for successively searching recordable areas dispersed on a record medium will be described using the area PRA. Referring to Fig. 29, the volume of the FRA is 03h. In this case, as shown in Fig. 29, the slot 03h is accessed. The start address and the end address recorded in the slot 03h represent the start address and the end address of one part on the disc.

[0008]

As shown in Fig. 29A, link information recorded in the slot 03h represents that the next slot address is 18h. Thus, as shown in Fig. 29B. the slot 18h is accessed. Link information recorded in the slot

18h represents that the next slot address is 1Fh. Likewise, as shown in Fig. 29C, the slot 1Fh is accessed. As shown in Fig. 29D, corresponding to link information in slot 1Fh, a slot 2Bh is accessed. As shown in Fig. 29E, corresponding to link information in the slot 2Bh, a slot E3h is accessed. In such a manner, link information is traced until a null (00h) is detected as link information. Thus, the addresses of recordable areas dispersed on the MD are successively recognized. Alternatively, by controlling an optical pickup and successively accessing these addresses, recordable areas dispersed in the memory can be obtained. Likewise, by referencing the P-DFA or the P-TNoN, defective areas that are dispersed in the memory can be successfully accessed.

[0009]

[Problem to be solved by the Invention]

Link-P system is possibly performing to control with above-described four parameters likewise of FAT. However, the configuration of the software is so complicated for managing it.

[0010]

The other hand, FAT file system of standard type PC is introduced to the memory card as the resulting of having the concept of the memory for the PC. Hence, the system is easily become a larger size, the management of FAT file system with the small CPU

for the memory is troublesome.

[0011]

With the Link-P system and FAT file system, when edit process of the data file such as a combine process for combining two data into one data, a divide process for dividing one data into two data or the like are performed, the file management can be easily performed.

[0012]

An object of the present invention is to provide an editing apparatus and an editing method for a nonvolatile memory for adding an attribute file to the beginning of each data file and manage parts that disperse in the memory with the attribute file so as to allow the editing process to be performed even if the FAT area is destroyed.

[0013]

[Means for Solving the Problem]

According to the present invention of claim 1, there is provided a data processing apparatus having an attachable/detachable non-volatile file memory from which data is readable by a computer, comprising:

a controlling portion for handling a data file and file management information for managing the data file; and

a memory interface disposed between said controlling portion and the non-volatile memory,

wherein the file management information is composed of:

a first file information area containing at least the file name of the data file and the number of parts that compose the data file, the number of parts being at least one; and

a second file information area containing at least the size of each of the part, the part start position, and the part end position, and

wherein when the data file is divided into two data files, the data processing apparatus is adapted for updating the size and the file end position of the second file information area corresponding to the data file so as to generate the file management information corresponding to a first data file placed before a divide point of the data file and for copying a predetermined area containing the divide point and placing the copied predetermined area at the beginning of the second data file so as to generate the file management information corresponding to a second data file placed after the dividing point.

[0014]

According to the present invention of claim 2, there is provided a data processing apparatus having an attachable/detachable non-volatile file memory from which data is readable by a computer, comprising:

a controlling portion for handling a data

file and file management information for managing the data file; and

a memory interface disposed between said controlling portion and the non-volatile memory,

wherein the file management information is composed of:

a first file information area containing at least the file name of the data file and the number of parts that compose the data file, the number of parts being at least one; and

a second file information area containing at least the size of each of the part, the part start position, and the part end position, and

wherein when a first data file and a second data file are combined, the data processing apparatus is adapted for successively placing the first file information area and the second file information area corresponding to the first data file and the second file information corresponding to the second data file so as to generate the file management information corresponding to the combined data file.

[0015]

According to the present invention of claim 3, there is provided a data processing method for a data processing apparatus having an attachable/detachable non-volatile file memory from which data is readable by a computer,

the non-volatile memory storing a data file and file management information for managing the data file,

the file management information being composed of a first file information area containing at least the file name of the data file and the number of parts that compose the data file, the number of parts being at least one, and a second file information area containing at least the size of each of the part, the part start position, and the part end position,

wherein the data file is divided into two data files, the data processing method comprising the steps of:

updating the size and the file end position of the second file information area corresponding to the data file so as to generate the file management information corresponding to a first data file placed before a divide point of the data file; and

copying a predetermined area containing the divide point and placing the copied predetermined area at the beginning of the second data file so as to generate the file management information corresponding to a second data file placed after the dividing point.

[0016]

According to the present invention of claim 4, there is provided a data processing method for a data processing apparatus having an

attachable/detachable non-volatile file memory from which data is readable by a computer, the non-volatile memory storing a data file and file management information for managing the data file, the file management information being composed of a first file information area containing at least the file name of the data file and the number of parts that compose the data file, the number of parts being at least one, and a second file information area containing at least the size of each of the part, the part start position, and the part end position,

wherein a first data file and a second data file are combined, the data processing method comprising the step of:

successively placing the first file information area and the second file information area corresponding to the first data file and the second file information corresponding to the second data file so as to generate the file management information corresponding to the combined data file.

[0017]

In the case of the editing processes such as combine like track A and B into track A for the data file which is recorded to the detachable non-volatile memory are performed, the part information area PRTINF of track B is moved after the part information area PRTINF of track A that had been moved, then the track

information area TRKINF is deleted. At this moment, afterward a chain of the sound file of track A is moved, a chain of the sound file of track B is also moved. Then, the track information area TRKINF of track A being updated and two of the part information area PRTINF are closely arranged. That is, the track information area TRKINF of track A, the part information area PRTINF of track A and the part information area PRTINF are arranged sequently.

[0018]

[Embodiment of the Invention]

Next, an embodiment of the present invention will be described. Fig. 1 is a block diagram showing the whole structure of according to an embodiment of the present invention. The digital audio recorder/player records and reproduces a digital audio signal using a detachable memory card. In reality, the recorder/player composes an audio system along with an amplifying unit, a speaker, a CD player, an MD recorder, a tuner, and so forth. However, it should be noted that the present invention can be applied to other audio recorders. In other words, the present invention can be applied to a portable recording/reproducing apparatus. In addition, the present invention can be applied to a set top box that records a digital audio data that is circulated as a satellite data communication, a digital broadcast, or

Internet. Moreover, the present invention can be applied to a system that records/reproduces moving picture data and still picture data rather than audio data. The system according to the embodiment of the present invention can record and reproduce additional information such as picture and text other than a digital audio signal.

[0019]

The recording/reproducing apparatus has an audio encoder/decoder IC 10, a security IC 20, a DSP (Digital Signal Processor) 30. Each of these devices is composed of a one-chip IC. The recording/reproducing apparatus has a detachable memory card 40. The one-chip IC of the memory card 40 has flash memory (non-volatile memory), a memory control block, and a security block. The security block has a DES (Data Encryption Standard) encrypting circuit. According to the embodiment, the recording/reproducing apparatus may use a microcomputer instead of the DSP 30.

[0020]

The audio encoder/decoder IC 10 has an audio interface 11 and an encoder/decoder block 12. The encoder/decoder block 12 encodes a digital audio data corresponding to a highly efficient encoding method and writes the encoded data to the memory card 40. In addition, the encoder/decoder block 12 decodes encoded

data that is read from the memory card 40. As the highly efficient encoding method, the ATRAC3 format that is a modification of the ATRAC (Adaptive Transform Acoustic Coding) format used in Mini-Disc is used.

[0021]

In the ATRAC3 format, audio data sampled at 44.1 kHz and quantized with 16 bits is highly efficiently encoded. In the ATRAC3 format, the minimum data unit of audio data that is processed is a sound unit (SU). 1 SU is data of which data of 1024 samples (1024 x 16 bits x 2 channels) is compressed to data of several hundred bytes. The duration of 1 SU is around 23 msec. In the highly efficient encoding method, the data amount of audio data is compressed to data that is around 10 times smaller than that of original data. As with the ATRAC1 format used in Mini-Disc, the audio signal compressed and decompressed corresponding to the ATRAC3 format less deteriorates in the audio quality.

[0022]

A line input selector 13 selectively supplies the reproduction output signal of an MD, the output signal of a tuner, or a reproduction output signal of a tape to an A/D converter 14. The A/D converter 14 converts the input line signal to a digital audio signal (sampling frequency = 44.1 kHz; the number of quantizing bits = 16). A digital input selector 16 selectively supplies a digital output signal of an MD,

a CD, or a CS (Satellite Digital Broadcast) to a digital input receiver 17. The digital input signal is transmitted through for example an optical cable. An output signal of the digital input receiver 17 is supplied to a sampling rate converter 15. The sampling rate converter 15 converts the digital input signal into a digital audio signal (sampling frequency = 44.1 kHz; the number of quantizing bits = 16).

[0023]

The encoder/decoder block 12 of the audio encoder/decoder IC 10 supplies encoded data to a DES encrypting circuit 22 through an interface 21 of the security IC 20. The DES encrypting circuit 22 has a FIFO 23. The DES encrypting circuit 22 is disposed so as to protect the copyright of contents. The memory card 40 also has a DES encrypting circuit. The DES encrypting circuit 22 of the recording/reproducing apparatus has a plurality of master keys and an apparatus-unique storage key. The DES encrypting circuit 22 also has a random number generating circuit. The DES encrypting circuit 22 can share an authenticating process and a session key with the memory card 40 that has the DES encrypting circuit. In addition, the DES encrypting circuit 22 can re-encrypt data with the storage key of the DES encrypting circuit.

[0024]

The encrypted audio data that is output from the DES encrypting circuit 22 is supplied to a DSP 30. The DSP 30 communicates with the memory card 40 through an interface. In this example, the memory card 40 is attached to an attaching/detaching mechanism (not shown) of the recording/reproducing apparatus. The DSP 30 writes the encrypted data to the flash memory of the memory card 40. The encrypted data is serially transmitted between the DSP 30 and the memory card 40. In addition, an external SRAM (Static Random Access Memory) 31 is connected to the DSP 30.

[0025]

A bus interface 32 is connected to the DSP 30. Data is supplied from an external controller (not shown) to the DSP 30 through a bus 33. The external controller controls all operations of the audio system. The external controller supplies data such as a record command or a reproduction command that is generated corresponding to a user's operation through an operation portion to the DSP 30 through the bus interface 32. In addition, the external controller supplies additional information such as image information and character information to the DSP 30 through the bus interface 32. The bus 33 is a bidirectional communication path. Additional information that is read from the memory card 40 is supplied to the external controller through the DSP

30, the bus interface 32, and the bus 33. In reality, the external controller is disposed in for example an amplifying unit of the audio system. In addition, the external controller causes a display portion to display additional information, the operation state of the recorder, and so forth. The display portion is shared by the audio system. Since data that is exchanged through the bus 33 is not copyright protected data, it is not encrypted.

[0026]

The encrypted audio data that is read from the memory card 40 by the DSP 30 is decrypted by the security IC 20. The audio encoder/decoder IC 10 decodes the encoded data corresponding to the ATRAC3 format. Output data of the audio encoder/decoder 10 is supplied to a D/A converter 18. The D/A converter 18 converts the output data of the audio encoder/decoder 10 into an analog signal. The analog audio signal is supplied to a line output terminal 19.

[0027]

The analog audio signal is supplied to an amplifying unit (not shown) through the line output terminal 19. The analog audio signal is reproduced from a speaker or a head set. The external controller supplies a muting signal to the D/A converter 18. When the muting signal represents a mute-on state, the external controller prohibits the audio signal from

being output from the line output terminal 19.

[0028]

Fig. 2 is a block diagram showing the internal structure of the DSP 30. Referring to Fig. 2, the DSP 30 comprises a core 34, a flash memory 35, an SRAM 36, a bus interface 37, a memory card interface 38, and inter-bus bridges. The DSP 30 has the same function as a microcomputer. The core 34 is equivalent to a CPU. The flash memory 35 stores a program that causes the DSP 30 to perform predetermined processes. The SRAM 36 and the external SRAM 31 are used as a RAM of the recording/reproducing apparatus.

[0029]

The DSP 30 controls a writing process for writing encrypted audio data and additional information to the memory card 40 corresponding to an operation signal such as a record command received through the bus interfaces 32 and 37 and a reading process for reading them therefrom. In other words, the DSP 30 is disposed between the application software side of the audio system that records/reproduces audio data and additional information and the memory card 40. The DSP 30 is operated when the memory card 40 is accessed. In addition, the DSP 30 is operated corresponding to software such as a file system.

[0030]

The DSP 30 manages files stored in the memory

card 40 with the FAT system used in conventional personal computers. In addition to the file system, according to the embodiment of the present invention, a management file is used. The management file will be descried later. The management file is used to manage data files stored in the memory card 40. The management file as the first file management information is used to manage audio data files. On the other hand, the FAT as the second file management information is used to mange all files including audio data files and management files stored in the flash memory of the memory card 40. The track information management file is stored in the memory card 40. The FAT is written to the flash memory along with the route directory and so forth before the memory card 40 is shipped. The details of the FAT will be described later.

[0031]

According to the embodiment of the present invention, to protect the copyright of data, audio data that has been compressed corresponding to the ATRAC3 format is encrypted. On the other hand, since there are not necessary to protect the copyright of an additional information and the track information management file, they are not encrypted. There are two types of memory cards that are an encryption type and a non-encryption type. However, a memory card for use

with the recorder/player that records copyright protected data is limited to the encryption type. Voice data and image data that are recorded by users are recorded on non-encryption type memory cards.

[0032]

Fig. 3 is a block diagram showing the internal structure of the memory card 40. The memory card 40 comprises a control block 41 and a flash memory 42 that are structured as a one-chip IC. A bidirectional serial interface is disposed between the DSP 30 of the recorder/player and the memory card 40. The bidirectional serial interface is composed of ten lines that are a clock line SCK for transmitting a clock signal that is transmitted along with data, a status line SBS for transmitting a signal that represents a status, a data line DIO for transmitting data, an interrupt line INT, two GND lines, two INT lines, and two reserved lines.

[0033]

The clock line SCK is used for transmitting a clock signal in synchronization with data. The status line SBS is used for transmitting a signal that represents the status of the memory card 40. The data line DIO is used for inputting and outputting a command and encrypted audio data. The interrupt line INT is used for transmitting an interrupt signal that causes the memory card 40 to interrupt the DSP 30 of the

recorder/player. When the memory card 40 is attached to the recorder/player, the memory card 40 generates the interrupt signal. However, according to the embodiment of the present invention, since the interrupt signal is transmitted through the data line DIO, the interrupt line INT is grounded.

[0034]

A serial/parallel converting, parallel/serial converting, and interface block (S/P, P/S, I/F block) 43 is an interface disposed between the DSP 30 of the recorder/player and the control block 41 of the memory card 40. The S/P, P/S, and IF block 43 converts serial data received from the DSP 30 of the recorder/player into parallel data and supplies the parallel data to the control block 41. In addition, the S/P, P/S, and IF block 43 converts parallel data received from the control block 41 into serial data and supplies the serial data to the DSP 30. When the S/P, P/S, and IF block 43 receives a command and data through the data line DIO, the S/P, P/S, and IF block 43 separates them into these that are normally accessed to the flash memory 42 and those that are encrypted.

[0035]

In the format of which data is transmitted through the data line DIO, after a command is transmitted, data is transmitted. The S/P, P/S, and IF block 43 detects the code of a command and determines

whether the command and data are those that are normally accessed or those that are encoded. Corresponding to the determined result, the S/P, P/S, and IF block 43 stores a command that is normally accessed to a command register 44 and stores data that is normally accessed to a page buffer 45 and a write register 46. In association with the write register 46, the memory card 40 has an error correction code encoding circuit 47. The error correction code encoding circuit 47 generates a redundant code that is an error correction code for data temporarily stored in the page buffer 45.

[0036]

Output data of the command register 44, the page buffer 45, the write register 46, and the error correction code encoding circuit 47 is supplied to a flash memory interface and sequencer (hereinafter, referred to as memory I/F and sequencer) 51. The memory IF and sequencer 51 is an interface disposed between the control block 41 and the flash memory 42 and controls data exchanged therebetween. Data is written to the flash memory through the memory IF and sequencer 51.

[0037]

Audio data that has been compressed corresponding to the ATRAC3 format and written to the flash memory (hereinafter, this audio data is referred

to as ATRAC3 data) is encrypted by the security IC 20 of the recorder/player and the security block 52 of the memory card 40 so as to protect the copyright of the ATRAC3 data. The security block 52 comprises a buffer memory 53, a DES encrypting circuit 54, and a non-volatile memory 55.

[0038]

The security block 52 of the memory card 40 has a plurality of authentication keys and a unique storage key for each memory card. The non-volatile memory 55 stores a key necessary for encrypting data. The key stored in the non-volatile memory 55 cannot be analyzed. According to the embodiment, for example, a storage key is stored in the non-volatile memory 55. The security block 52 also has a random number generating circuit. The security block 52 authenticates an applicable recorder/player and shares a session key therewith. In addition, the security block 52 re-encrypts contents with the storage key through the DSE encrypting circuit 54.

[0039]

For example, when the memory card 40 is attached to the recorder/player, they are mutually authenticated. The security IC 20 of the recorder/player and the security block 52 of the memory card 40 mutually authenticate. When the recorder/player has authenticated the attached memory

card 40 as an applicable memory card and the memory card 40 has authenticated the recorder/player as an applicable recorder/player, they are mutually authenticated. After the mutual authenticating process has been successfully performed, the recorder/player and the memory card 40 generate respective session keys and share them with each other. Whenever the recorder/player and the memory card 40 authenticate each other, they generate respective session keys.

[0040]

When contents are written to the memory card 40, the recorder/player encrypts a contents key with a session key and supplies the encrypted data to the memory card 40. The memory card 40 decrypts the contents key with the session key, re-encrypts the contents key with a storage key, and supplies the contents key to the recorder/player. The storage key is a unique key for each memory card 40. When the recorder/player receives the encrypted contents key, the recorder/player performs a formatting process for the encrypted contents key, and writes the encrypted contents key and the encrypted contents to the memory card 40.

[0041]

In the above section, the writing process for the memory card 40 was described. In the following, the reading process for the memory card 40 will be

described. Data that is read from the flash memory 42 is supplied to the page buffer 45, the read register 48, and the error correction circuit 49 through the memory IF and the sequencer 51. The error correcting circuit 49 corrects an error of the data stored in the page buffer 45. Output data of the page buffer 45 that has been error-corrected and the output data of the read register 48 are supplied to the S/P, P/S, and IF block 43. The output data of the S/P, P/S, and IF block 43 is supplied to the DSP 30 of the recorder/player through the above-described serial interface.

[0042]

When data is read from the memory card 40, the contents key encrypted with the storage key and the contents encrypted with the block key are read from the flash memory 42. The security block 52 decrypts the contents key with the storage key. The security block 52 re-encrypts the decrypted content key with the session key and transmits the re-encrypted contents key to the recorder/player. The recorder/player decrypts the contents key with the received session key and generates a block key with the decrypted contents key. The recorder/player successively decrypts the encrypted ATRAC3 data.

[0043]

A configuration ROM 50 is a memory that

stores partition information, various types of attribute information, and so forth of the memory card 40. The memory card 40 also has an erase protection switch 60. When the switch 60 is in the erase protection position, even if a command that causes the memory card 40 to erase data stored in the flash memory 42 is supplied from the recorder/player side to the memory card 40, the memory card 40 is prohibited from erasing the data stored in the flash memory 42. An OSC cont. 61 is an oscillator that generates a clock signal that is the reference of the timing of the process of the memory card 40.

[0044]

Fig. 4 is a schematic diagram showing the hierarchy of the processes of the file system of the computer system that uses a memory card as a storage medium. On the hierarchy, the top hierarchical level is an application process layer. The application process layer is followed by a file management process layer, a logical address management layer, a physical address management layer, and a flash memory access layer. In the above-mentioned hierarchical structure, the file management process layer is the FAT file system. Physical addresses are assigned to individual blocks of the flash memory. The relation between the blocks of the flash memory and the physical addresses thereof does not vary. Logical addresses are addresses

that are logically handled on the file management process layer.

[0045]

Fig. 5 is a schematic diagram showing the physical structure of data handled in the flash memory 42 of the memory card 40. In the memory 42, a data unit (referred to as segment) is divided into a predetermined number of blocks (fixed length). One block is divided into a predetermined number of pages (fixed length). In the flash memory, data is erased as each block at a time. Data is written to the flash memory 42 or read therefrom as a page at a time. The size of each block is the same. Likewise, the size of each page is the same. One block is composed of page 0 to page m. For example, one block has a storage capacity of for example 8 KB (kilobytes) or 16 KB. One page has a storage capacity of 512 B (bytes). When one block has a storage capacity of 8 KB, the total storage capacity of the flash memory 42 is 4 MB (512 blocks) or 8 MB (1024 blocks). When one block has a storage capacity of 16 KB, the total storage capacity of the flash memory 42 is 16 MB (1024 blocks), 32 MB (2048 blocks), or 64 MB (4096 blocks).

[0046]

One page is composed of a data portion of 512 bytes and a redundant portion of 16 bytes. The first three bytes of the redundant portion is an overwrite

portion that is rewritten whenever data is updated. The first three bytes successively contain a block status area, a page status area, and an update status area. The remaining 13 bytes of the redundant portion are fixed data that depends on the contents of the data portion. The 13 bytes contain a management flag area (1 byte), a logical address area (2 bytes), a format reserve area (5 bytes), a dispersion information ECC area (2 bytes), and a data ECC area (3 bytes). The dispersion information ECC area contains redundant data for an error correction process against the management flag area, the logical address area, and the format reserve area. The data ECC area contains redundant data for an error correction process against 512-byte data.

[0047]

The management flag area contains a system flag (1: user block, 0: boot block), a conversion table flag (1: invalid, 0: table block), a copy prohibition flag (1: OK, 0: NG), and an access permission flag (1: free, 0: read protect).

[0048]

The first two blocks - blocks 0 and 1 are boot blocks. The block 1 is a backup of the block 0. The boot blocks are top blocks that are valid in the memory card. When the memory card is attached to the recorder/player, the boot blocks are accessed at first.

The remaining blocks are user blocks. Page 0 of the boot block contains a header area, a system entry area, and a boot and attribute information area. Page 1 of the boot block contains a prohibited block data area. Page 2 of the boot block contains a CIS (Card Information Structure)/IDI (identify Drive Information) area.

[0049]

The header area of the boot block contains a boot block ID and the number of effective entries. The system entries are the start position of prohibited block data, the data size thereof, the data type thereof, the data start position of the CIS/IDI area, the data size thereof, and the data type thereof. The boot and attribute information contains the memory card type (read only type, rewritable type, or hybrid type), the block size, the number of blocks, the number of total blocks, the security/non-security type, the card fabrication data (date of fabrication), and so forth.

[0050]

Since the flash memory has a restriction for the number of rewrite times due to the deterioration of the insulation film, it is necessary to prevent the same storage area (block) from being concentratedly accessed. Thus, when data at a particular logical address stored at a particular physical address is rewritten, updated data of a particular block is

written to a non-used block rather than the original block. Thus, after data is updated, the relation between the logical address and the physical address changes. This process is referred to as swap process. Consequently, the same block is prevented from being concentratedly accessed. Thus, the service life of the flash memory can be prolonged.

[0051]

The logical address associates with data written to the block. Even if the block of the original data is different from the block of updated data, the address on the FAT does not change. Thus, the same data can be properly accessed. However, since the swap process is performed, a conversion table that correlates logical addresses and physical addresses is required (this table is referred to as logical-physical address conversion table). With reference to the logical-physical address conversion table, a physical address corresponding to a logical address designated on the FAT is obtained. Thus, a block designated with a physical address can be accessed.

[0052]

The DSP 30 stores the logical-physical address conversion table in the SRAM. When the storage capacity of the RAM is small, the logical-physical address conversion table can be stored to the flash memory. The logical-physical address conversion table

correlates logical addresses (2 bytes) sorted in the ascending order with physical addresses (2 bytes). Since the maximum storage capacity of the flash memory is 128 MB (8192 blocks), 8192 addresses can be assigned with two bytes. The logical-physical address conversion table is managed for each segment. Thus, the size of the logical-physical address conversion table is proportional to the storage capacity of the flash memory. When the storage capacity of the flash memory is 8 MB (two segments), two pages are used as the logical-physical address conversion table for each of the segments. When the conversion table is stored in the flash memory, a predetermined one bit of the management flag area in the redundant portion in each page represents whether or not the current block is a block containing the logical-physical address conversion table.

[0053]

The above-described memory card can be used with the FAT file system of a personal computer system as with the disc shaped record medium. The flash memory has an IPL area, a FAT area, and a route directory area (not shown in Fig. 5). The IPL area contains the address of a program to be initially loaded to the memory of the recorder/player. In addition, the IPL area contains various types of memory information. The FAT area contains information with

respect to blocks (clusters). The FAT has defined unused blocks, next block number, defective blocks, and last block number. The route directory area contains directory entries that are a file attribute, an update date [day, month, year], file size, and so forth.

[0054]

According to the embodiment of the present invention, in addition to the file management system defined in the format of the memory card 40, the file management information (the track information management file) is specified corresponding to music files. The management file is recorded to a user block of the flash memory 42 of the memory card 40. Thus, as will be described later, even if the FAT of the memory card 40 is destroyed, a file can be recovered.

[0055]

The track information management file is generated by the DSP 30. When the power of the recorder/player is turned on, the DSP 30 determines whether or not the memory card 40 has been attached to the recorder/player. When the memory card has been attached, the DSP 30 authenticates the memory card 40. When the DSP 30 has successfully authenticated the memory card 40, the DSP 30 reads the boot block of the flash memory 42. Thus, the DSP 30 reads the physical-logical address conversion table and stores the read data to the SRAM. The FAT and the route directory have

been written to the flash memory of the memory card 40 before the memory card 40 is shipped. When data is recorded to the memory card 40, the track information management file is generated.

[0056]

In other words, a record command issued by the remote controller of the user or the like is supplied to the DSP 30 from the external controller through the bus and the bus interface 32. The encoder/decoder IC 10 compresses the received audio data and supplies the resultant ATRAC3 data to the security IC 20. The security IC 20 encrypts the ATRAC3 data. The encrypted ATRAC3 data is recorded to the flash memory 42 of the memory card 40. Thereafter, the FAT and the track information management file are updated. Whenever a file is updated (in reality, whenever the recording process of audio data is completed), the FAT and the track information management file stored in the SRAMs 31 and 36 are rewritten. When the memory card 40 is detached or the power of the recorder/player is turned off, the FAT and the track information management file that are finally supplied from the SRAMs 31 and 36 are recorded to the flash memory 42. Alternatively, whenever the recording process of audio data is completed, the FAT and the management file written in the flash memory 42 may be rewritten. When audio data is edited, the contents of

the track information management file are updated.

[0057]

In the data structure according to the embodiment, an additional information management file is generated. An additional information management file is updated and recorded to the flash memory 42 that is performed as well as the track information management file. The additional information is supplied from the external controller to the DSP 30 through the bus and the bus interface 32. The additional information is recorded to the flash memory 42 of the memory card 40. Since the additional information is not supplied to the security IC 20, it is not encrypted. When the memory card 40 is detached from the recorder/player or the power thereof is turned off, the additional information management file is written from the SRAM of the DSP 30 to the flash memory 42. An additional information management file on the flash memory 42 may be updated by each time of recording an additional information.

[0058]

Fig. 6 is a schematic diagram showing the file structure of the memory card 40. As the file structure, there are a still picture directory, a moving picture directory, a voice directory, a control directory, and a music directory. According to the embodiment, music programs are recorded and reproduced.

Next, the music directory will be described. The music directory consists of a track information management file TRKLIST.MSF (hereinafter, referred to as TRKLIST), a backup track information management file TRKLISTB.MSF (hereinafter, referred to as TRKLISTB), an additional information file INFLIST.MSF (that contains an artist name, an ISRC code, a time stamp, a still picture data, and so forth (this file is referred to as INFIST)), an ATRAC3 data file A3Dnnnn.MSF (hereinafter, referred to as A3nnnn). The area NAME1 is an area that contains the memory card name and the program name (for one byte code corresponding to ASCII/8859-1 character code). The area NAME2 is an area that contains the memory card name and the program name (for two byte code corresponding to MS-JIS/Hankul/Chinese code).

[0059]

Fig. 7 is a schematic diagram showing the relation between the track information management file TRKLIST, the areas NAME1 and NAME2, and the ATRAC3 data file A3Dnnnn. The file TRKLIST is a fixed-length file of 64k bytes (= 16 k x 4). An area of 32k bytes of the file is used for managing tracks. The remaining area of 32k bytes is used to contain the areas NAME1 and NAME2. Although the areas NAME1 and NAME2 for program names may be provided as a different file as the track information management file, in a system having a small storage capacity, it is convenient to

totally manage the track information management file and program name files.

[0060]

The track information management file TRKINF-nnnn and part information area PRTINF-nnnn of the track information management file TRKLIST are used to manage the data file A3Dnnnn and the additional information INFLIST. Only the ATRAC3 data file A3Dnnnn is encrypted. In Fig. 6, the data length in the horizontal direction is 16 bytes (0 to F). A hexadecimal number in the vertical direction represents the value at the beginning of the current line.

[0061]

Next, with reference to Fig. 8, the relation betmusic programs and ATRAC3 data files will be described. One track is equivalent to one music program. The numbers of the tracks which are able to record in the memory card are maximumly limited to 400 tracks. In addition, one music program is composed of one ATRAC3 data (see Fig. 8). The ATRAC3 data file is audio data that has been compressed corresponding to the ATRAC3 format. The ATRAC3 data file is recorded as a cluster at a time to the memory card 40. One cluster has a capacity of 16 KB. A plurality of files are not contained in one cluster. The minimum data erase unit of the flash memory 42 is one block. In the case of the memory card 40 for music data, a block is a synonym

of a cluster. In addition, one cluster is equivalent to one sector.

[0062]

One music program is basically composed of one part. However, when a music program is edited, one music program may be composed of a plurality of parts. A part is a unit of data that is successively recorded in the track information management file TRLIST.MSF. A unit of the data which is recorded from the start time of recording to the stop time of recording that constitutes the part. Normally, one track is composed of one part. There is the limitation with the maximum value of part. The number of parts are P and the number of tracks are T (1 to 400), there is the relation between the parts and tracks (P=2043-4XT) which can be used. For example, when one track is composed of 2039 parts, the part is impossibly assigned for the second music that leads the difficulty with making the file for the second music.

[0063]

SU is the minimum unit of a part. In addition, SU is the minimum data unit in the case that audio data is compressed corresponding to the ATRAC3 format. 1 SU is audio data of which data of 1024 samples at 44.1 kHz (1024 x 16 bits x 2 channels) is compressed to data that is around 10 times smaller than that of original data. The duration of 1 SU is around

23 msec. Normally, one part is composed of several thousand SU.

[0064]

Fig. 8 is a schematic diagram showing the file structure in the case that two music programs of a CD or the like are successively recorded. The first program (file 1) is composed of for example five clusters. Since one cluster cannot contain two files of the first program and the second program, the file 2 starts from the beginning of the next cluster. Thus, the end of the part 1 corresponding to the file 1 is in the middle of one cluster and the remaining area of the cluster contains no data. Likewise, the second music program (file 2) is composed of one part.

[0065]

There are four types of edit processes that are a divide process, a combine process, an erase process, and a move process. The divide process is performed to divide one track into two portions. When the divide process is performed, the number of total tracks increases by one. In the divide process, one file is divided into two files on the file system. Thus, in this case, the track information management file TRKLIST.MSF is updated. The erase process is performed to erase a track. The track numbers after the track that has been erased decrease one by one. The move process is performed to change the track

sequence and other meaning of the move is the process that moves other record medium such as a hard disk fro the memory card not the internal of the memory card. The copy is performed to create the copy of the original. Opposing to it, the move only has the meaning of the movement. Accordingly, the copy of the original music is hardly generated by the move.

[0066]

Fig. 8 is a schematic diagram showing the combined result of two programs (file 1 and file 2) shown in Fig. 9. As a result of the combine process, the combined file is composed of two parts. Fig. 10 is a schematic diagram showing the divided result of which one program (file 1) is divided in the middle of the cluster 2. By the divide process, the file 1 is composed of clusters 0, 1, and the beginning portion of cluster 2. The file 2 is composed of the end portion of cluster 2 and clusters 3 and 4.

[0067]

When an above-described of edit process is performed, rewriting the data file of ATRAC3 takes a longer time. The file management information TRKLIST.MSF which contains the edit points are only rewritten. Consequently, the concept of the part is introduced for this purpose.

[0068]

Fig. 11 is a schematic diagram showing the

detailed data structure of the reproduction management file PBLIST. The size of the reproduction management file is one cluster (one block = 16 KB). Also, TRKLISTB which follows TRLIST has the same size and data as PBLIST. At the track information management file, the size of the header is from the top of TRLIST.MSF to 32 bytes which can be represented as (0x0000) and (0x0010). Furthermore, in the middle of the file, the unit which delimits the header by 8 bytes from the top of TRLIST.MSF is referred as slot. In the case of the track information management file, 16 bytes are referred to as slots. The following data for providing first slot of the file that supplies in sequence from the top of the header.

[0069]

BLK ID-TL0/TL1 (4 bytes)

Fixed value (for example, TL0 = 0x544C2D30, TL1 = 0x544C2D31)

T-TRK (2 bytes)

The selected numbers of music (1 to 400)

MCode (2 bytes)

The recorder maker and the marker code for identifying types.

The management code for specifying the recorder maker which recorded the memory card that is given from the licensor at the time of licensing. Those type codes are managed by each of licensees.

REVISION (4 bytes)

Number of rewrite times of PBLIST.MSF

Increments whenever the reproduction management file is rewritten.

S-YMDhms (4 bytes)

Year, month and day of TRKLIST.MSF are lately updated.

N1 (OP) (one byte)

The sequential number for the memory card (a molecular side), when one sheet is used, most of memory cards are (0x01). (OP) stands for optional.

N2 (OP) (one byte)

The sequential number for the memory card (denominator side), when one sheet is used, the most of memory cards are (0x01).

MSID (OP) (2 byte)

It is ID for the memory card. When it is a purality of the combination, MSID are the same number (T.B.D.). (T.B.D. has the meaning of determing in the future.)

S-TRK (2 bytes)

The record (T.B.D.) of the special track (401 to 408) that is normally 0x0000.

PASS (OP) (2 bytes)

Password (T.B.D.)

APP (OP) (2 bytes)

The specification of the reproduction

application (T.B.D.) (Normally, it is 0x0000.)

INF-S (OP) (2 bytes)

It is the additional information pointer for the entire memory card, When the addtional information is not contained, it is normally 00.

S_YMDhms (OP) (4 bytes)

Year, month and day of TRKLIST.MSF are updated by the reliable equipment for recording the time accurately.

[0070]

BLK ID-TL0 which is the same BLK ID-TL0 in the internal header, MCode and REVISION are provided as the last 16 bytes of TRKLIST.MSF. Thus, the header is written into the backup version of TRKLISTB.MSF. In this case, BLK ID-TL1, MCode and REVISION are provided.

[0071]

There is any possibility of pulling out the memory card during the time of recording and the power goes off, when the audio equipment is recovered, the detections of the errors are required as the conventional type of audio equipment. In the meantioning above, REVISION is written into the header and the end of the block and incremented plus 1 by each time of rewriting the values. In this case, the track information management file has backup that allows easy-recovery of the last status. There is four REVISION which includes backup which realizes higher

percentage of the error detection. When the error detection is finished, the alarms such as the error massage or the like are generated.

[0072]

In addition, since the fixed value BLK ID-TL0/TL1 is written at the beginning of one block (16 KB), when the FAT is destroyed, the fixed value is used as a reference for recovering data. In other words, with reference to the fixed value, the type of the file can be determined. Since the fixed value BLK ID-TL0/TL1 are redundantly written at the header and the end portion of each block, the reliability can be secured.

[0073]

The data amount of an ATRAC3 data file is much larger than that of the track information management file. In addition, as will be described later, a block number BLOCK SERIAL is added to ATRAC3 data file. However, since a plurality of ATRAC3 files are recorded to the memory card, to prevent them from become redundant, both CONNUM0 and BLOCK SERIAL are used. Otherwise, when the FAT is destroyed, it will be difficult to recover the file.

[0074]

Likewise, the maker code (MCode) is redundantly recorded at the beginning and the end of each block so as to identify the maker and the model in

such a case that a file has been improperly recorded in the state that the FAT has not been destroyed.

[0075]

The header is followed by a track information area TRKINF for information with respect to each track and a part information area PRTINF for information with respect to each part of tracks (music programs). Fig. 11 shows the areas preceded by the area TRKLIST. The lower portion of the area TRKLISTB shows the detailed structure of these areas. In Fig. 11, a hatched area represents an unused area. The following information is recorded in the track information area TRKINF-nnn that will be described in sequence.

[0076]

T0 (1 byte)

Fixed value (T0 = 0x74)

LT (one byte)

LT (1 byte)

Either the limitation with the reproduction is existing or not. (0x80: There is the limitation with the reproduction, 0x00: There is no limitation with the reproduction, Others: Stop the reproduction).

INF-nnn (Option) (2 bytes)

Represents the additional information pointer (0 to 409) of each track. 00: music program without additional information.

FNM-nnn (4 bytes)

Represents the file number (0x0000 to 0xFFFF) of an ATRK3 data file.

The number nnnnn (in ASCII) of the ATRAC3 data file name (A3Dnnnn.MSA) is converted into 0xnnnnn.

CONTENTS KEY (8 bytes)

The special value which is created by each of the contents that is encrypted in the security block of the memory card.

[0077]

S-SAM (D) SERIAL (16 bytes)

Represents the serial number unique to the machine that has the memory card.

APP_CTL (OP) (4 bytes)

Represents the application parameter (T.B.D.). (Normally, 0x0000)

CONNUM (4 bytes)

Represents the contents cumulation number.

CONNUM is generated for each contents file and uniquely stored in the security block of the recorder.

P-nnn (2 bytes)

Represents the number of parts that compose one music program (1 to 2039).

XT (OP) (2 bytes)

Represents the reproduction duration (SU) starting from the point represented by INX. 0000: not set, FFFF: up to end of music program

INX-nnn (OP) (2 bytes)

Represents a pointer of a particular portion of a music program (the beginning portion of a featured portion). INX-nnn is designated with the number of relative SU starting from the beginning of the music program. The music scan function is improved so that the user can designate a featured portion unlike with the conventional function that allows the user to listen to the each music program for only 10 seconds from the beginning thereof.

YMDhms-S (4 bytes)

Represents the reproduction start date and time for a reproduction restricted track. 0x00000000: when not used

YMDhms-E (4 bytes)

Represents the reproduction expiration date and time for a reproduction restricted track. 0x00000000: when not used

MT (1 byte)

Represents the number of times of reproduction permission for a reproduction restricted track. 0x00: when not used.

CT (1 byte)

Represents the number of times of reproduction for a reproduction restricted track. 0x00: when not used.

CC (1 byte)

Represents a copy control byte. 00: copy prohibited, 01: first copy generation, 10: copy free. A child file of a first copy generation file is copy-prohibited.

CN (1 byte)

Represents the number of times of copy operation. 00: copy prohibited. 01 to 0xFE: number of times of copy operation. 0xFF: number of times of copy operation not restricted. Valid for a first generation copy file. CN is counted up whenever copy operation is performed.

[0078]

Part information area PRTINF-nnn contains part information of the track. Next, the part information will be described in the order of the arrangement.

[0079]

PR (1 byte)

Fixed value (PR = 0x50)

A-nnnn (2 bytes)

Represents part attribute information containing mode (1 byte) and SCMS (Serial Copy Management System) information (1 byte).

PRTSIZE-nnnn (4 bytes)

Represents a part cluster size (2 bytes), a start SU (1 byte), and an end SU (1 byte).

PRTKEY-nnnn (8 bytes)

Paired with a contents key for creating a block key for encrypting music data. The initial value of PRTKEY-nnnn is 0. Whenever a part is generated in the editing operation, the value of PRTKEY-nnnn is incremented by +1.

[0080]

Fig. 12 shows mode information of the ATRAC3 mode represented with the low order byte of A-nnnn. The mode information is defined as shown in Fig. 12. Fig. 12 shows the number of bytes, record duration (for a 64 MB memory card), data transmission rate, and compression rate for six modes of HQ, SP, CD, LP1, LP2, and monaural.

[0081]

Fig. 13 shows information represented with the high order byte of A-nnnn. Bit 0 represents emphasis on/off information. Bit 1 represents reproduction skip/normal reproduction information. Bit 2 represents data type (for example, audio data or data sound such as FAX). Bits 3 and 4 are reserved. With a combination of bits 5 and 6, as shown in Fig. 13, SCMS information is defined. Bit 7 represents write permission/prohibition information.

[0082]

Fig. 14 shows the detailed data structure of NAME1 (area for one byte code). Each of NAME1 and NAME2 (that will be described later) is divided every

eight bytes (= 1 slot) from the beginning. At the beginning (first slot) 0x8000 of NAME1, the following header is placed. The header (first slot 0x8008) is followed by pointers and names. At the last slot of NAME1, the same data as the header is placed.

[0083]

BLK ID-NM1 (4 bytes)

Fixed value representing the contents of the block (NM1 = 0x4E4D2D31).

MCode (2 bytes)

Code identifying a company and a machine type.

[0084]

PNM1-nnn (OP) (4 bytes)

Represents a pointer to NM1 (1-byte code)

PNM1-S represents a pointer to the representative name of the memory card.

nnn (= 1 to 408) represents a pointer of a music program name.

The pointer represents the start position of the block (2 bytes), character code type (2 bits), and data size (14 bits).

The start position represents a byte offset value starting from the beginning of the NM1 area (0x000 to 0x3989).

The character code type represents:

0: ASCII. 1: ASCII + kana. 2: modified

8859-1.

The data size (14 bits) represents the total value of character data and one byte at the end (0x00) (0x000 to 0x398C).

NM1-nnn (OP)

One-byte code, representing a memory card name and program name data in variable length.

End code of name data (0x00) is placed.

[0085]

Fig. 15 shows the detailed data structure of NAME2 (area for 2-byte code). At the beginning (the first slot) (0x8000) of NAME2, the following header is placed. The header is followed by pointers and names. The same data as the header is placed at the last slot of NAME2.

[0086]

BLK ID-NM2 (4 bytes)

Fixed value representing the contents of the block (NM2 = 0x4E4D2D32).

MCode (2 bytes)

Code identifying a company and a machine type.

[0087]

PNM2-nnn (OP) (4 bytes)

Represents a pointer to NM2 (2-byte code).

PNM2-S represents a pointer for the representative name of the memory card.

nnn (= 1 to 408) represents a pointer for a music program name.

The pointer represents the start position of the block (2 bytes), the character code type (2 bits), and the data size (14 bits).

The start position represents a byte offset value starting from the beginning of the NM2 area (0x000 to 0x3987).

The character code type represents:

0: Japanese (MS-JIS). 1: Korean (KS C5601-1989).

2: Chinese (GB2312-80).

The data size (14 bits) represents the total value of the character data and two bytes at the end (0x0000) (0x000 to 0x398C).

NM2-nnn (OP)

2-byte code, representing a memory card name and a music program name in variable length.

End code of name data (0x0000) is placed.

[0088]

Fig. 16 shows a data arrangement (for one block) of an ATRAC3 data file A3Dnnnn.MSA in the case that 1 SU is composed of N bytes. In this file, one slot is composed of eight bytes. In Fig. 15, the beginning and end portions of the file are shown as slots (0x0000 to 0x3FF8). The first four slots of the file are used for a header. The header contains the

following data. In the second last slot (0x3FF0), BLOCK SPEED is redundantly placed. In the last slot (0x3FF8), BLK ID-A3D and MCode are redundantly placed.

[0089]

BLK ID-A3D (4 bytes)

Fixed value that identifies the contents of the block (A3D = 0x41324420).

MCode (2 bytes)

Code, identifying a company and a machine type. When the file is edited, MCode should be rewritten.

BLOCK SPEED (8 bytes)

Used to create a block key for encrypting a file. The start value of the block seed is a random number calculated by the security block of the recorder. BLOCK SEED of the next block is incremented by +1. For countermeasures against an error, the same data is redundantly written at the beginning and end of the block. Even if the file is edited, it is not necessary to rewrite BLOCK SEED.

CONNUM (4 bytes)

Represents the contents number obtained at first. The same value as CONNUM of TRKLIST.MSF. Even if the file is edited, it is not necessary to rewrite CONNUM.

BLOCK SERIAL (4 bytes)

BLOCK SERIAL of the first block is 0. BLOCK

SERIAL of the next block is incremented by +1. Even if the file is edited, it is not necessary to rewrite BLOCK SERIAL.

INITIALIZATION VECTOR (8 bytes)

Represents the initial value required when each block of ATRAC3 data is encrypted or decrypted. The value of INITIALIZATION VECTOR at the beginning of the contents is 0. The value of INITIALIZATION VECTOR of the next block is the last encrypted value of the last SU. Even if the file is edited, it is not necessary to rewrite INITIALIZATION VECTOR.

[0090]

The header is followed by sound unit data SU-nnnn. SU is data of which 1024 samples are compressed. The data amount of 1 SU depends on each mode. Even if the file is edited, it is not necessary to rewrite SU-nnnn. Fig. 17 shows the data amount of SU, the number of SU per block, the reserved data amount per block, the transmission rate, and the duration of each mode.

[0091]

For example, consider the case that a 64 MB memory card is used in CD mode. The 64 MB memory card has 3968 blocks. In the CD mode, since 1 SU is composed of 320 bytes, one block has 51 SU. 1 SU is equivalent to (1024 / 44100) seconds. Thus, one block is equivalent to (1024 / 44100) x 51 x (3968 - 16) = 4680 seconds = 78 minutes. The transmission rate is

(44100 / 1024) x 320 x 8 = 110250 bps.

[0092]

Fig. 18 shows the detailed data structure of an additional information management file INFLIST.MSF that contains additional information. Since the file INFLIST.MSF is a part of the track information management file TRKLIST.MSF, the file is divided every 16 bytes (= one slot) from the beginning. At the beginning (first slot) (0x0000), the following header is placed. The header is followed by pointers and data.

[0093]

BLK ID-INF (4 bytes)

Fixed value identifying the contents of the block (INF = 0x494E464F).

T-DAT (2 bytes)

Represents the total number of data elements (0 to 409).

MCode (2 bytes)

Represents the maker code of the recorder.

YMDhms (1 byte)

Represents the record update date and time.

INF-nnn (4 bytes)

Represents a pointer to additional information DATA (variable length, every 2 bytes (slot)). The start position is represented with the high order 16 bits of INF-nnn (0000 to FFFF). INF-nnn

represents the offset value from the beginning (0x0800) of Data Slot-0000 (for each slot). The data size is represented with the low order 16 bits of INF-nnn (0001 to 7FFF). A valid/invalid flag is set at the most significant bit MSB. MSB = 0 (valid). MSB = 1 (invalid).

The data size represents the total amount of data. (The data starts from the beginning of each slot. At the end of the data, 0 is written until the end of the slot).

The first INF represents a pointer for additional information of the entire album (normally, INF-409).

[0094]

Fig. 19 shows the data structure of additional information. At the beginning of the additional information, a header of eight bits is added. The header contains the following data. The header is followed by variable length data.

[0095]

IN (1 byte)

Fixed value. (IN = 0x69)

ID (1 byte)

Represents a large category of additional information. ID is referred to as key ID against sub ID.

SID (1 byte)

Represents the type of sub ID (T.B.D.).

SIZE (2 bytes)

Represents the size of addition information of each ID for each slot (1 to 7FFF). A valid/invalid flag is set at the most significant bit MSB. MSB = 0 (valid). MSB = 1 (invalid).

MCode (2 bytes)

Represents the maker code of the recorder.

[0096]

Fig. 20 shows an example of additional information. When the size is 0x8xxx, erased data or invalid data is represented. The additional information is distinguished with codes of the header (for example, key ID and SID). However, since these values (codes) have not been defined, they are not shown. The additional information includes copyright code ISRC (International Standard Recording Code), music program information such as composer name and artist name, and hardware control information. In the case of music program information, a character code is added to the first two bytes of the data.

[0097]

Fig. 21 is showing the structure of one additional information. Some examples of the additional information are described here. The additional information indicates the case of the time stamp shown in Fig. 22. The time stamp is the time

stamp for recording as shown in Fig. 20. The data is YMHhms. 00 is written into the area with remaining of 1 slot. Fig. 23 shows the additional information for the reproduction log file. Year, month, day (YMD), and minutes (hms) are written by the log data.

[0098]

Fig. 24 shows the additional information for the artist names, ISRC code, and TOCID. In this case, the artist names are recorded by using of one byte code. 00 is written into the remaining of the slot. ISRC code is written as the data in the following slot. Besides, the data of TOC-ID is written afterwards of the slot. When the additional information that indicates in Fig. 24 is deleted, and the additional information of Fig. 24 is rewritten to the additional information of Fig. 25. In other words, the size is 8xxx.

[0099]

According to the second embodiment of the present invention, in addition to the file system defined as a format of the memory card, the track information management file TRKLIST. or music data is used. Thus, even if the FAT is destroyed, the file can be recovered. Fig. 26 shows a flow of a file recovering process. To recover the file, a computer that operates with a file recovery program and that can access the memory card (The computer has a function

equivalent to the DSP30.) and a storing device (hard disk, RAM, or the like) connected to the computer are used. At the first step 101, the following process will be performed.

[0100]

All blocks of the flash memory whose FAT has been destroyed are searched for TL-0 as the value (BLKID) at the top position of each block. In addition, all the blocks are searched for NM-1 as the value (BLKID) at the top position of each block. Thereafter, all the blocks are searched for NM-2 as the value (BLKID) at the top position of each block. All the contents of the four blocks (track information management file) are stored to, for example a hard disk by the recovery computer.

[0101]

The number of total tracks is obtained from data after the fourth byte of the track information management file. The 20-th byte of the track information area TRKINF-001, the value of the area CONNUM-001 of the first music program, and the value of the next area P-001 are obtained. The number of parts is obtained with the value of the area P-001. The values of the areas PRTSIZE of all parts of the track 1 of the area PRTINF is obtained. The number of total blocks (clusters) n is calculated and obtained.

[0102]

After the track information management file is obtained, the flow advances to step 102. At step 102, a voice data file (ATRAC3 data file) is searched. All blocks of other than the management file is searched from the flash memory. Blocks whose top value (BLKID) is A3D are collected.

[0103]

A block of which the value of the area CONNUM0 at the 16-th byte of A3Dnnnn is the same as that of the area CONNUM-001 of the first music program of the track information management file and of which the value of the area BLOCK SERIAL that starts from 20-th byte is 0 is searched. After the first block is obtained, a block (cluster) with the same value of the area CONNUM value as the first block and of which the value of BLOCK SERIAL is incremented by 1 (1 = 0 + 1) is searched. After the second block is obtained, a block with the same value of the area CONNUM0 as the second block and of which the value of the area BLOCK SERIAL is incremented by 1 (2 = 1 + 1) is searched.

[0104]

By repeating the process, the ATRC3 data file is searched until n blocks (clusters) of the track 1 are obtained. When all the blocks (clusters) are obtained, they are successively stored to the hard disk.

[0105]

The same process for the track 1 is performed for the track 2. In other words, a block of which the value of the area CONNUM0 is the same as that of the area CONNUM-002 of the first music program of the track information management file and of which the value of the area BLOCK SERIAL that starts at the 20-th byte is searched. Thereafter, in the same manner as the track 1, the ATRAC3 data file is searched until the last block (cluster) n' is detected. After all blocks (clusters) are obtained, they are successively stored to the hard disk.

[0106]

By repeating the above-described process for all tracks (the number of tracks: m), all the ATRAC3 data is stored to the hard disk controlled by the recovering computer.

[0107]

At step 103, the memory card whose the FAT has been destroyed is re-initialized and then the FAT is reconstructed. A predetermined directory is formed in the memory card. Thereafter, the track information management file and the ATRAC3 data file for m tracks are copied from the hard disk to the memory card. Thus, the recovery process is finished.

[0108]

In the management file and data file, important parameters (in particular, codes in headers)

may be recorded triply rather than doubly. When data is redundantly recorded, the same data may be recorded at any positions as long as they are apart from each other for one page or more.

[0109]

In this section, a combine process for combining two tracks A and B composed of one part each will be described by using the flow chart of Fig. 40. At step 201, the part information area PRTINF of the track B on the backward side is moved below the part information area PRTINF of the track A. Thus, in the track information management file TRKLIST, the track information area TRKINF of the track A, the part information area PRTINF of the track A, the part information area PRTINF of the track B, and the track information area TRKINF of the track B are successively arranged.

[0110]

At step 202, the chain of the FAT of the ATRAC3 data file of the track B is linked on the backward side of the chain of the FAT of the ATRAC3 data file of the track A. At step 203, the track information area TRKINF of the track B is deleted from the track information management file TRKLIST. Thus, in the track information management file TRKLIST, the track information area TRKINF of the track A, the part information area PRTINF of the track A, and the part

information area PRTINF of the track B are successively arranged. At step 204, the ATRAC3 data file of the track B is deleted from the directory. At step 205, P-nnn that represents the number of parts that compose a music program in the track information area TRKINF of the track A is changed from 1 to 1 + 1 = 2.

[0111]

Thus, the values of the keys are changed. In this example, the contents key of the original track A is denoted by KC_A; and the contents key of the original track B is denoted by KC_B. Likewise, the part key of the original track A is denoted by KP_A; and the part key of the original track B is denoted by KP_B.

[0112]

At step 206, after the tracks A and B are combined, the contents key of the new track N is generated as KC_N. The CONNUM is also newly generated. At step 207, a new part key is generated. The new part key is generated with an exclusive OR operation of the contents key KC_A, the part key KP_A, and the contents key KC_N. At step 408, the backward-side part key (namely, the part key for the part information area PRTINF of the original track B) is generated. As with the new part key, the backward key on the backward side is generated with an exclusive OR operation of the contents key KC_B, the part key KP_B, and the contents

key KC_N.

[0113]

At step 209, the contents key KC_N of the new track N is encrypted with the storage key of the memory card and stored in the CONTENTS KEY-nnn of the track information area TRKINF. The CONNUM is stored in the CONNUM-nnn of the track information area TRKINF. In addition, each part key is stored in the PRTKEY-nnn of the part information area PRTINF.

[0114]

Next, with reference to Fig. 35, the divide process (see Fig. 10C) for dividing a track A composed of one part into two tracks A and B will be described. Fig. 35 is a flow chart showing the divide process. At step 301, the divide point is decided with SU. At step 302, the PRTSIZE of the part information area PRTINF of the new track A is changed. In reality, the number of clusters from the beginning (start SU) to the divide point (end SU) is counted. The cluster size, the start SU, and the end SU are changed corresponding to the position of SU of the divide point of the cluster and stored to the PRTSIZE of the part information area of the new track A.

[0115]

At step 303, one cluster that is the last cluster of the new track A that contains the divide point is completely copied. The copied cluster is

treated as the top part of the new track B. At step 304, the number of total parts of the newly generated track B is stored to the P-nnn representing the number of parts that compose a music program in the track information area TRKINF of the track B. In this example, clusters preceded by the divide point become the second part that is the newly generated track B. The number of total parts of the newly generated track B is counted. At step 305, the file number FNW-nnn of the new ATRAC3 data file is generated and stored to the FNW-nnn of the track information area TRKINF.

[0116]

At step 306, the track information area TRKINF of the new track B and the part information area PRTINF are added on the backward side of the part information area PRTINF of the new track A of the track information management file TRKLIST. The track information area TRKINF of the track on the backward side of the original track A and the part information area PRTINF are moved backward by the track information area TRKINF and the part information area PRTINF of the track B.

[0117]

At step 307, the chain of the FAT of the ATRAC3 data file of the new track A is changed to the divide point. At step 308, since the track B is newly added, the file B of the ATRAC3 data file is added to

the directory. At step 309, the chain of the FAT of the ATRAC3 data file of the newly generated track B is followed by the remaining portion of the original track A (namely, the chain of the clusters including the divide point).

[0118]

Since the new track B is added, the key values are added. However, the key values of the new track A are not changed.

[0119]

At step 310, after the track is divided, the contents key KC_B of the contents key of the new track B is generated. In addition, the CONNUM is newly generated. At step 311, the part key KP_B of the new track B is generated. The part key of the new track is generated with an exclusive OR operation of the original KC_A, KP_A, and KC_B.

[0120]

At step 312, the contents key KC_B of the new track B is encrypted wit the storage key of the memory card and stored to the CONTENTS KEY-nnn of the track information area TRKINF. In addition, the CONNUM is stored to the CONNUM-nnn of the track information area TRKINF. Each part key is directly stored to the PRTKEY-nnn of the part information area PRTINF.

[0121]

Thus, even if edit processes such as combine

and divide processes are performed, the track information area TRKINF and the part information area PRTINF are arranged in the same order of the sound file as the content of the track information management file TRKLIST.MSF. In other words, unlike with the Link-P system, the track information area TRKINF of one file that has been edited and the link destination of the part information area PRTINF are arranged continuously, not randomly.

[0122]

In addition, when another edit process such as erase process or move process is performed, the track information area TRKINF and the part information area PRTINF are arranged.

[0123]

Likewise, when edit process for the data file which is recorded in the memory card is performed, the embodiment of the present invention is solved the problem with link P that is adopted as the simple method for MD which arranges the track information area TRKINF in which the information of the data file is recorded and the part information area PRTINF for the part in which is recorded in the track information area TRKINF.

[0124]

[Effect of the Invention]

According to this invention, the data file is

managed by FAT file system but, clusters and sound units which consist the data file are managed by part unit, edit process for example, combine, divide, erase and move corresponding to the data file which is recorded in the memory card are performed that allows managing the small CPU for the memory easily.

[0125]

The flash memory has an individual unit like page, when edit process such as combine, divide, erase and move for the data file which is recorded in the memory card is performed, problem with the random position of link P is used for MD which requires many pages of access are solved.

[Brief Description of the Drawings]

[Fig. 1]

Block diagram showing the entire structure according to the present invention.

[Fig. 2]

Block diagram showing the internal structure of a DSP according to the present invention.

[Fig. 3]

Block diagram showing the internal structure of a memory card according to the present invention.

[Fig. 4]

Schematic diagram showing the structure of a file system processing hierarchical in the flash memory according to the present invention.

[Fig. 5]

Schematic diagram showing the physical structure of data in a flash memory according to the present invention.

[Fig. 6]

Schematic diagram showing the file convention according to the present invention.

[Fig. 7]

Schematic diagram showing the relation between files according to the present invention.

[Fig. 8]

Schematic diagram showing the data structure of the data file according to the present invention.

[Fig. 9]

Schematic diagram showing the example of edit processing of the data file according to the present invention.

[Fig. 10]

Schematic diagram showing other example of edit processing of the data file according to the present invention.

[Fig. 11]

Schematic diagram showing the structure of the track information management file.

[Fig. 12]

Schematic diagram showing the convention of a part attribute information in the track information

management file.

[Fig. 13]

Schematic diagram showing the convention of a part attribute information in the track information management file.

[Fig. 14]

Schematic diagram showing the structure of a name file in the track information management file for other data structure.

[Fig. 15]

Schematic diagram showing the structure of a name file in the track information management file for other data structure.

[Fig. 16]

Schematic diagram showing the data structure of data file according to the present invention..

[Fig. 17]

Schematic diagram showing types of recording mode and recording hours or the like for each recording mode according to the present invention.

[Fig. 18]

Schematic diagram showing the structure of an additional information management file according to the present invention.

[Fig. 19]

Schematic diagram showing the structure of an additional information data according to the present

invention.

[Fig. 20]

Schematic diagram showing the example of the additional information according to the present invention.

[Fig. 21]

Schematic diagram showing the structure of an additional information according to the present invention.

[Fig. 22]

Schematic diagram showing the case of the structure for an additional information is the time stamp according to the present invention.

[Fig. 23]

Schematic diagram showing the case of the structure for an additional information is the reproduction log file according to the present invention.

[Fig. 24]

Schematic diagram showing the case of the structure for an additional information is the artist names according to the present invention.

[Fig. 25]

Schematic diagram showing the case of the structure for an additional information is the artist names that are deleted according to the present invention.

[Fig. 26]

Schematic diagram explaining the flow of file recovery process.

[Fig. 27]

Flow chart explaining the example of the combine process according to the invention.

[Fig. 28]

Flow chart explaining the example of the divide process according to the invention.

[Fig.29]

Schematic diagram explaining the data file for MD.

[Description of Reference Numerals]

10 ... Audio encoder/decoder IC, 20 ... Security IC, 30 ... DSP, 40 ... Memory card, 42 ... Flash memory, 52 ... Security block, TRKLIST.MSF ... Track information management file, INFLIST.MSF ... Additional information management file, A3Dnnn.MSA ... Audio data file

[Title of Document] Abstract

[Abstract]

[Subject]

Edit process of the file is performed for managing the small CPU for the memory easily.

[Solving means]

In the case of the editing processes such as combine like track A and B into track A for the data file which is recorded to the detachable non-volatile memory are performed, the part information area PRTINF of track B is moved after the part information area PRTINF of track A that had been moved, then the track information area TRKINF is deleted. At this moment, afterward a chain of the sound file of track A is moved, a chain of the sound file of track B is also moved. When the divide is performed and the divide point of the cluster is copied, then TRKINF and PRTINF are updated by determining the first half of the chain to the divide point as track A, TRKINF and PRTINF are generated by determing the divide point to the second half of the chain as track B. TRKINF and PRTINF are moved from the original position to the new place as the portion of TRKINF and PRTINF for new track B.

[Selected Drawing] Fig. 11